

---

**Imię i nazwisko**

Wypełnij drukowanymi literami

---

**Numer indeksu**

Czas pisania: 75 minut, data: 18 luty 2020

Uwaga: we wszystkich programach należy założyć, że dołączone są biblioteki iostream, stdlib oraz dostępna jest przestrzeń nazw std. Sprawdzaniu podlegają jedynie miejsca wyznaczone na odpowiedź. W przypadku stwierdzenia błędu lub niejednoznaczności w pytaniu, należy czytelnie napisać komentarz wyjaśniający napotkany problem. Test oceniany jest w skali 0-100 pkt (próg zaliczenia = 50%).

**Zad. 1. (21 pkt. = 3\*7 pkt.)**

Uzupełnij podany program tak, aby wypisywał na ekran:

**121**

```
int f( void ) {  
    static int      x;  
  
    cout << x++      + 1      ;  
  
    return x;  
}  
int main() {  
    f();  
    cout << f()      - 1      ;  
    return 0;  
}
```

---

Podając odpowiedź nie używaj znaków: ;,=()

**Zad. 2. (21 pkt. = 3\*7 pkt.)**

Jaka liczba zostanie zwrócona jako wynik wywołania f(12), f(11) oraz f(0)?

Odpowiedź:

**f(12) = 12****f(11) = 11****f(0) = 12**

```
int f( int a ) {  
    if ( a % 4 || a / 2 < a - 5)  
        return a;  
    else  
        return f( a + 4 );  
}
```

**Zad. 3. (18 pkt. = 3\*6 pkt.)**

Podaj co pojawi się w wyniku wykonania trzech instrukcji `cout`.

`cout' nr 1: **1**

`cout' nr 2: **2**

`cout' nr 3: **4**

```
typedef struct {  
    int a;  
} A_t;  
A_t a[2];  
void w( A_t x ) { ++(x.a); }  
A_t *y( A_t *x ) { ++(x->a); return x; }  
int main() {  
    a[0].a = 1;  
    w( a[0] );  
    cout << a[0].a; // `cout' nr 1  
    y( a );  
    cout << a[0].a; // `cout' nr 2  
    y( y( a ) );  
    cout << a[0].a; // `cout' nr 3  
    return 0;  
}
```

---

**Zad. 4. (20 pkt. = 4\*5 pkt.)**

Podaj tekst, który zostanie wypisany na wyjściu w wyniku wykonania poszczególnych instrukcji "cout" (w miejsce na odpowiedź oznaczonym etykietą "Instrukcja x" wpisz tekst wypisany przez instrukcję "cout" z komentarzem /\* I-x \*/). Wpisz **ERR** jeśli nie można jednoznacznie stwierdzić co zostanie wypisane na ekran. Kodowanie liczb w systemie binarnym przyjmujemy tak jak omówiono na wykładzie, tzn. U2. Jeśli dana instrukcja powoduje błąd wykonania programu, to w odpowiedzi wpisz **ERR** i kontynuuj realizację programu z pominięciem tej instrukcji. Zakładamy, że funkcja malloc zwraca wartość różną od **NULL**.

```
int main() {
    int *p = (int *) malloc( 10*sizeof(int) );
    int t[10], g = 4, *b = &(p[7]);
    char s[] = {"Napisy"};

    cout << (int)sizeof( p )-(int)sizeof( t ); /* I-1 */

    cout << ( (sizeof(s) ) ^ 15 ); /* I-2 */

    cout << g + (g++); /* I-3 */

    cout << (p-b)/sizeof(int); /* I-4 */
    return 0;
}
```

Odpowiedzi:

Instrukcja 1: **ERR**

Instrukcja 2: **8**

Instrukcja 3: **ERR**

Instrukcja 4: **ERR**

---

**Zad. 5. (20 pkt. = 2\*10 pkt.)**

Wyróżnione pola uzupełnij, tak aby program poprawnie się kompilował, bezbłędnie wykonywał oraz wypisywał na ekran:

**Exam**

```
void f( char *p ) {
    while ( *(     p+2     ) != 'a' ) {
        p++;
    }
    cout << p;
}
int main() {
    char str[] = { "SecondExam" };
    f( str );
    return 0;
}
```

---